

WHAT IS CLAIMED IS:

1. An image forming apparatus, comprising:
a program obtaining part for sending a screen
5 data for inputting a program to a client terminal, and
receiving the program from the client terminal; and
a program execution part for executing the
program received by the program obtaining part.

10

2. An image forming apparatus, comprising:
an operation display part for displaying a
15 screen and inputting information;
a program obtaining part for displaying a
screen on the operation display part for inputting a
program and obtaining the program input from the
operation display part; and
20 a program execution part for executing the
program obtained by the program obtaining part.

25

3. The image forming apparatus as claimed in
claim 1, wherein the program obtaining part includes a
Web server capability, and the image forming apparatus
sends a html file as the screen data to the client
5 terminal according to access from the client terminal.

10 4. The image forming apparatus as claimed in
claim 3, wherein the image forming apparatus detects an
event from the client terminal and executes a CGI
program corresponding to the event.

15

5. The image forming apparatus as claimed in
claim 1, wherein, when the program is executed in the
20 image forming apparatus, the image forming apparatus
causes the client terminal to display a screen same as
a screen displayed on a display part of the image
forming apparatus.

25

6. The image forming apparatus as claimed in
claim 1, wherein the image forming apparatus associates
5 a key with the program according to a request from the
client terminal or an operation display part of the
image forming apparatus, and stores a table including
correspondence between the program and the key.

10

7. The image forming apparatus as claimed in
claim 6, wherein the image forming apparatus detects
15 input of the key, and executes the program
corresponding to the key by referring to the table.

20

8. The image forming apparatus as claimed in
claim 6, the image forming apparatus displays a key
assigning screen for assigning a key to the program,
and assigns a key to a program according to input from
25 the key assigning screen.

5 9. The image forming apparatus as claimed in
claim 1, wherein the image forming apparatus executes
the program for testing operation of the program.

10

10. The image forming apparatus as claimed in
claim 1, the image forming apparatus further comprising
a part for accessing a server that stores programs and
15 downloading a program from the server.

20 11. The image forming apparatus as claimed in
claim 1, wherein the program execution part is an
interpreter.

25

12. The image forming apparatus as claimed in
claim 1, the image forming apparatus further
comprising:

5 hardware resources used for image forming
processing;

control services existing between the
hardware resources and an application in the image
forming apparatus;

10 wherein the program operates as the
application.

15

13. The image forming apparatus as claimed in
claim 12, wherein the program performs function call to
the control services by using the program execution
part.

20

14. A program execution method in an image
25 forming apparatus, the program execution method

comprising the steps of:

 sending a screen data for inputting a program
 to a client terminal;

5 receiving the program from the client
 terminal; and

 executing the program received from the
 client terminal.

10

15. A program execution method in an image
 forming apparatus, the program execution method
 comprising the steps of:

15 displaying a screen on an operation display
 part in the image forming apparatus for inputting a
 program;

 obtaining the program input from the
 operation display part; and

20 executing the program obtained by the program
 obtaining part.

25

16. The program execution method as claimed
in claim 14, wherein the image forming apparatus sends
a html file as the screen data to the client terminal
according to access from the client terminal.

5

17. The program execution method as claimed
10 in claim 16, wherein the image forming apparatus
detects an event from the client terminal and executes
a CGI program corresponding to the event.

15

18. The program execution method as claimed
in claim 14, wherein, when the program is executed in
the image forming apparatus, the image forming
20 apparatus causes the client terminal to display a
screen same as a screen displayed on a display part of
the image forming apparatus.

25

19. The program execution method as claimed
in claim 14, wherein the image forming apparatus
associates a key with the program according to a
5 request from the client terminal or an operation
display part of the image forming apparatus, and stores
a table including correspondence between the program
and the key.

10

20. The program execution method as claimed
in claim 19, wherein the image forming apparatus
15 detects input of the key, and executes the program
corresponding to the key by referring to the table.

20

21. The program execution method as claimed
in claim 19, the image forming apparatus displays a key
assigning screen for assigning a key to the program,
and assigns a key to a program according to input from
25 the key assigning screen.

5 22. The program execution method as claimed
in claim 14, the image forming apparatus comprising:
 hardware resources used for image forming
processing;
 control services existing between the
10 hardware resources and an application in the image
forming apparatus;
 wherein the program operates as the
application.

15

 23. A program causing an image forming
apparatus to obtain an application program, the program
20 comprising:
 program obtaining program code means for
sending a screen data for inputting an application
program to a client terminal, and receiving the
application program from the client terminal.

25

24. A program for casing an image forming
5 apparatus to obtain an application program, wherein the
image forming apparatus comprising an operation display
part for displaying a screen and inputting information,
the program comprising:

program obtaining program code means for
10 displaying a screen on the operation display part for
inputting an application program and obtaining the
application program input from the operation display
part.

15

25. The program as claimed in claim 23,
wherein the program obtaining program code means
20 includes a Web server capability, and causes the image
forming apparatus to send a html file as the screen
data to the client terminal according to access from
the client terminal.

25

26. The program as claimed in claim 23, the
program further comprising program code means for
5 causing the client terminal to display a screen same as
a screen displayed on a display part of the image
forming apparatus when the application program is
executed in the image forming apparatus.

10

27. The program as claimed in claim 23, the
program further comprising program code means for
15 accessing a server that stores programs and downloading
a program from the server.

20

28. A computer readable recording medium
storing a program causing an image forming apparatus to
obtain an application program, the program comprising:
program obtaining program code means for
25 sending a screen data for inputting an application

program to a client terminal, and receiving the application program from the client terminal.

5

29. An information processing apparatus used for developing a program to be executed on an image forming apparatus, the information processing apparatus 10 comprising:

an execution part for interpreting and executing the program;

an emulator for causing the information processing apparatus to perform processing 15 corresponding to that to be performed by the image forming apparatus according to instructions of the program.

20

30. The information processing apparatus as claimed in claim 29, wherein the execution part is a virtual machine.

25

31. The information processing apparatus as
5 claimed in claim 30, the information processing
apparatus further comprising a compiler for converting
source code of the program to code executable by the
virtual machine.

10

32. The information processing apparatus as
claimed in claim 31, wherein the program is a Java
15 program and the information processing apparatus
further comprises a class library to be referred to by
the Java program.

20

33. The information processing apparatus as
claimed in claim 29, wherein the emulator causes the
information processing apparatus to display a screen
25 corresponding to an operation panel of the image

forming apparatus.

5

34. The information processing apparatus as claimed in claim 29, wherein the emulator causes the information processing apparatus to display a list of programs and execute a program selected by the user.

10

35. The information processing apparatus as 15 claimed in claim 29, wherein the emulator causes the information processing apparatus to print data to a printer connected to the information processing apparatus for a print instruction.

20

36. The information processing apparatus as claimed in claim 29, the information processing 25 apparatus further comprising a sending part for sending

a program.

5

37. The information processing apparatus as claimed in claim 36, wherein the sending part sends the program to a server that stores programs.

10

38. The information processing apparatus as claimed in claim 36, wherein the sending part receives 15 a screen data for inputting a program and sends the program to the image forming apparatus by using the screen data.

20

39. A program for casing an information processing apparatus to emulate an image forming apparatus, wherein the information processing apparatus 25 is used for developing an application program to be

executed on the image forming apparatus, the program comprising:

emulator program code means for causing the information processing apparatus to perform processing 5 corresponding to that to be performed by the image forming apparatus according to instructions of the application program.

10

40. The program as claimed in claim 39, the program further comprising execution program code means for interpreting and executing the application program.

15

41. The program as claimed in claim 40, 20 wherein the execution program code means is a virtual machine.

25

42. The program as claimed in claim 41, the program further comprising a compiler for converting source code of the application program to code executable by the virtual machine.

5

43. The program as claimed in claim 42,
10 wherein the application program is a Java program and the program further comprises a class library to be referred to by the Java program.

15

44. The program as claimed in claim 39,
wherein the emulator program code means causes the information processing apparatus to display a screen
20 corresponding to an operation panel of the image forming apparatus.

25

45. The program as claimed in claim 39,
wherein the emulator program code means causes the
information processing apparatus to display a list of
application programs and execute an application program
5 selected by the user.

10 46. The program as claimed in claim 39,
wherein the emulator program code means causes the
information processing apparatus to print data to a
printer connected to the information processing
apparatus according to a print instruction.

15

47. The program as claimed in claim 39, the
20 program further comprising sending program code means
for sending an application program.

25

48. The program as claimed in claim 47,
wherein the sending program code means causes the
information processing apparatus to send the
application program to a server that stores application
5 programs.

10 49. The program as claimed in claim 47,
wherein the sending program code means causes the
information processing apparatus to receive a screen
data for inputting an application program and sends the
application program to the image forming apparatus by
15 using the screen data.

20 50. A program, to be executed on an image
forming apparatus, developed by using an information
processing apparatus, the information processing
apparatus comprising:
an execution part for interpreting and
25 executing the program;

an emulator for causing the information processing apparatus to perform processing corresponding to that to be performed by the image forming apparatus according to instructions of the 5 program.

10 51. A computer readable medium storing a program for casing an information processing apparatus to emulate an image forming apparatus, wherein the information processing apparatus is used for developing an application program to be executed on the image 15 forming apparatus, the program comprising:

emulator program code means for causing the information processing apparatus to perform processing corresponding to that to be performed by the image forming apparatus according to instructions of the 20 application program.

25 52. An image forming apparatus that includes

service modules for performing system side processing on image formation, wherein applications can be added to the image forming apparatus separately from the service modules, the image forming apparatus

5 comprising:

a virtual machine for executing an application; and

an application management part for managing the application executed by the virtual machine.

10

53. The image forming apparatus as claimed in

15 claim 52, the image forming apparatus comprising an interpreter instead of the virtual machine.

20

54. The image forming apparatus as claimed in

claim 52, wherein the application is a Java program and the image forming apparatus includes a class library to be referred to by the Java program.

25

55. The image forming apparatus as claimed in
5 claim 54, wherein the class library includes classes
for displaying data on a display part of the image
forming apparatus.

10

56. The image forming apparatus as claimed in
claim 52, wherein the application management part is a
loader for loading the application into the image
15 forming apparatus from a recording medium to be
connected to the image forming apparatus or from a
server to be connected to the image forming apparatus
via a network.

20

57. The image forming apparatus as claimed
in claim 56, wherein the loader determines whether an
25 application is already loaded in the image forming

apparatus, executes the application if the application is loaded, and displays a loader screen if the application is not loaded.

5

58. The image forming apparatus as claimed in
claim 56, wherein the loader displays an application
10 load screen on a display part of the image forming
apparatus and downloads an application from a location
specified by the user on the load screen.

15

59. The image forming apparatus as claimed in
claim 58, wherein the loader displays a list of
applications and downloads an application selected from
20 the list by the user.

25

60. The image forming apparatus as claimed in

claim 58, wherein the location is a Web server or a FTP server that stores applications.

5

61. The image forming apparatus as claimed in
claim 58, wherein the loader checks, at predetermined
intervals, whether an application corresponding to the
10 downloaded application is updated in the location.

15 62. The image forming apparatus as claimed in
claim 61, wherein the loader displays a screen for
inputting the interval on a display part of the image
forming apparatus and sets the interval according to
input by the user.

20

63. The image forming apparatus as claimed in
25 claim 52, the image forming apparatus further

comprising:

a connecting part for connecting applications
and causing the image forming apparatus to perform a
series of processes according to the connected
5 applications.

10 64. The image forming apparatus as claimed in
claim 63, wherein the connecting part displays, on a
display part of the image forming apparatus, a first
application and a plurality of applications that are
candidates to be connected to the first application,
15 and connects an application selected from among the
plurality of applications to the first application.

20

65. A program to be executed in an image
forming apparatus that includes service modules for
performing system side processing on image formation,
wherein applications can be added to the image forming
25 apparatus separately from the service modules, the

program comprising:

virtual machine program code means for
executing an application; and
application management program code means for
5 managing the application executed by the virtual
machine program code means.

10

66. The program as claimed in claim 65, the
program comprising interpreter program code means for
interpreting and executing the application instead of
the virtual machine program code means.

15

67. The program as claimed in claim 65,
20 wherein the application is a Java program and the
program includes a class library to be referred to by
the Java program.

25

68. The program as claimed in claim 67,
wherein the class library includes classes for
displaying data on a display part of the image forming
5 apparatus.

10 69. The program as claimed in claim 65,
wherein the application management program code means
is loader program code means for loading the
application into the image forming apparatus from a
recording medium to be connected to the image forming
15 apparatus or from a server to be connected to the image
forming apparatus via a network.

20 70. The program as claimed in claim 65,
wherein the loader program code means determines
whether an application is already loaded in the image
forming apparatus, executes the application if the
25 application is loaded, and displays a loader screen if

the application is not loaded.

5

71. The program as claimed in claim 65,
wherein the loader program code means displays an
application load screen on a display part of the image
forming apparatus and downloads an application from a
10 location specified by the user.

15

72. The program as claimed in claim 71,
wherein the loader program code means displays a list
of applications on a display part of the image forming
apparatus and downloads an application selected from
the list by the user.

20

73. The program as claimed in claim 71,
25 wherein the location is a Web server or a FTP server

that stores applications.

5

74. The program as claimed in claim 71,
wherein the loader program code means checks, at
predetermined intervals, whether an application
corresponding to the downloaded application is updated
10 in the location.

15

75. The program as claimed in claim 74,
wherein the loader program code means displays a screen
for inputting the interval on a display part of the
image forming apparatus and sets the interval according
to input by the user.

20

76. The program as claimed in claim 65, the
25 program further comprising:

connecting program code means for connecting applications and causing the image forming apparatus to perform a series of processes according to the connected applications.

5

77. The program as claimed in claim 76,
10 wherein the connecting program code means displays, on a display part of the image forming apparatus, a first application and a plurality of applications that are candidates to be connected to the first application, and connects an application selected from among the 15 plurality of applications to the first application.

20 78. A computer readable medium storing a program to be executed in an image forming apparatus that includes service modules for performing system side processing on image formation, wherein applications can be added to the image forming 25 apparatus separately from the service modules, the

program comprising:

virtual machine program code means for
executing an application; and
application management program code means for
5 managing the application executed by the virtual
machine program code means.

10

79. An application execution method in an
image forming apparatus that includes a virtual machine
for executing an application and an application
management part for managing the application executed
15 by the virtual machine, the method comprising the steps
of:

displaying an application load screen on a
display part of the image forming apparatus;
downloading an application from a location
20 specified by the user; and
executing the application by using the
virtual machine.

25

80. The application execution method as
claimed in claim 79, wherein the image forming
apparatus includes an interpreter instead of the
5 virtual machine.

10 81. The application execution method as
claimed in claim 79, wherein the location is a
recording medium to be connected to the image forming
apparatus or a server to be connected to the image
forming apparatus via a network.

15

82. The application execution method as
20 claimed in claim 79, wherein the image forming
apparatus displays a list of applications and downloads
an application selected from the list by the user.

25

83. An information processing apparatus,
comprising a utility library that is implemented in a
program;

5 wherein the program includes nested modules
 in which each module is based on a state transition
 model,

 wherein, in the state transition model,
 operation of the program is determined according to a
10 first state of the program, an event for the program in
 the state, an event function to be executed when the
 event occurs in the state, and a second state
 transferred from the first state after the event
 function is executed,

15 wherein the utility library includes a state
 transition model launch function that assigns areas
 each for a module to a shared variable shared by the
 modules.

20

84. The information processing apparatus as
claimed in claim 83, wherein the each area is specified
25 by a pointer in a stack area.

5 85. The information processing apparatus as
claimed in claim 83, wherein the state transition model
10 launch function assigns areas each for a thread to a
 shared variable shared by threads.

10

15 86. The information processing apparatus as
claimed in claim 83, wherein the state transition model
 launch function executes an entry function that defines
 processing performed when state transition occurs in
 the state transition model.

20

25 87. The information processing apparatus as
claimed in claim 83, wherein the state transition model
 launch function executes a standard function that
 defines processing performed when an asynchronous event

that occurs irrespective of the state occurs.

5

88. The information processing apparatus as claimed in claim 83, wherein the information processing apparatus is an image forming apparatus.

10

89. The information processing apparatus as claimed in claim 88, the information processing apparatus further comprising:

hardware resources used for image forming processing;

control services that provide services to applications in the image forming apparatus, in which 20 the services includes control of hardware resources;

wherein the program is an application in the image forming apparatus, and the utility library is linked to the application.

25

90. A program producing method, comprising a
link step of linking a utility library with a program;
5 wherein the program includes nested modules
 in which each module is based on a state transition
 model,
 wherein, in the state transition model,
 operation of the program is determined according to a
10 first state of the program, an event for the program in
 the state, an event function to be executed when the
 event occurs in the state, and a second state
 transferred from the first state after the event
 function is executed,
15 wherein the utility library includes a state
 transition model launch function that assigns areas
 each for a module to a shared variable shared by the
 modules.

20

91. The method as claimed in claim 90,
wherein the each area is specified by a pointer in a
25 stack area.

5 92. The method as claimed in claim 90,
wherein the state transition model launch function
assigns areas each for a thread to a shared variable
shared by threads.

10

15 93. The method as claimed in claim 90,
wherein the state transition model launch function
executes an entry function that defines processing
performed when state transition occurs in the state
transition model.

20

25 94. The method as claimed in claim 90,
wherein the state transition model launch function
executes a standard function that defines processing
performed when an asynchronous event that occurs

irrespective of the state occurs.

5

95. The method as claimed in claim 90,
wherein the program produced according to the method is
a program to be executed in an image forming apparatus.

10

96. The method as claimed in claim 95, the information processing apparatus further comprising:

15

hardware resources used for image forming processing;

wherein the program is an application in the image forming apparatus

20 wherein the program is an application in the
image forming apparatus.

97. A state transition model in a program in
which operation of the program is determined according
to a first state of the program, an event for the
program in the state, an event function to be executed
5 when the event occurs in the state, and a second state
transferred from the first state after the event
function is executed,

wherein the state transition model is used in
an information processing apparatus including a utility
10 library that is implemented in the program;

wherein the program includes nested modules
in which each module is based on the state transition
model,

wherein the utility library includes a state
15 transition model launch function that assigns areas,
each of which areas is for a module, to a shared
variable shared by the modules.

20

98. A program produced by a program producing
method, the method comprising a link step of linking a
utility library with a program;

25 wherein the program includes nested modules

in which each module is based on a state transition model,

wherein, in the state transition model,
operation of the program is determined according to a
5 first state of the program, an event for the program in
the state, an event function to be executed when the
event occurs in the state, and a second state
transferred from the first state after the event
function is executed,
10 wherein the utility library includes a state
transition model launch function that assigns areas
each for a module to a shared variable shared by the
modules.

15

99. A program for causing an information
processing apparatus to generate nested modules in
20 which each module is based on a state transition model,
wherein, in the state transition model,
operation of the program is determined according to a
first state of the program, an event for the program in
the state, an event function to be executed when the
25 event occurs in the state, and a second state

transferred from the first state after the event
function is executed, the program comprising:

program code means that assigns areas each
for a module to a shared variable shared by the modules
5 in response to receiving a message; and

program code means for executing the event
function corresponding to the message by referring to
an event function table.

10

100. The program as claimed in claim 99, the
program further comprising program code means for
15 executing an entry function that defines processing
performed when state transition occurs in the state
transition model.

20

101. The program as claimed in claim 99, the
program further comprising program code means for
executing a standard function that defines processing
25 performed when an asynchronous event that occurs

irrespective of the state occurs.

5

102. The program as claimed in claim 99,
wherein the information processing apparatus is an
image forming apparatus.

10

103. The program as claimed in claim 102, the
information processing apparatus further comprising:

15 hardware resources used for image forming
processing;

 control services that provide services to
applications in the image forming apparatus, in which
the services includes control of hardware resources;

20 wherein the program is an application in the
image forming apparatus.

25

104. A computer readable medium storing a program produced by a program producing method, the method comprising a link step of linking a utility library with a program;

5 wherein the program includes nested modules in which each module is based on a state transition model,

 wherein, in the state transition model, operation of the program is determined according to a 10 first state of the program, an event for the program in the state, an event function to be executed when the event occurs in the state, and a second state transferred from the first state after the event function is executed,

15 wherein the utility library includes a state transition model launch function that assigns areas each for a module to a shared variable shared by the modules.

20

105. A computer readable medium storing a program for causing an information processing apparatus 25 to generate nested modules in which each module is

based on a state transition model,

wherein, in the state transition model,

operation of the program is determined according to a
first state of the program, an event for the program in

5 the state, an event function to be executed when the
event occurs in the state, and a second state
transferred from the first state after the event
function is executed, the program comprising:

program code means that assigns areas each
10 for a module to a shared variable shared by the modules
in response to receiving a message; and

program code means for executing the event
function corresponding to the message by referring to
an event function table.

15

20

25